

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A)

平1-216427

⑬ Int. Cl.<sup>4</sup>

G 06 F 9/38

識別記号

3 3 0

庁内整理番号

B-7361-5B

⑭ 公開 平成1年(1989)8月30日

審査請求 未請求 請求項の数 4 (全21頁)

⑮ 発明の名称 データ処理装置

⑯ 特 願 昭63-41086

⑰ 出 願 昭63(1988)2月24日

⑱ 発 明 者 上 田 達 也 兵庫県伊丹市瑞原4丁目1番地 三菱電機株式会社エル・エス・アイ研究所内

⑲ 発 明 者 吉 田 豊 彦 兵庫県伊丹市瑞原4丁目1番地 三菱電機株式会社エル・エス・アイ研究所内

⑳ 出 願 人 三菱電機株式会社 東京都千代田区丸の内2丁目2番3号

㉑ 代 理 人 弁理士 大岩 増雄 外2名

明 細 書

1. 発明の名称

データ処理装置

2. 特許請求の範囲

(1) 条件分岐命令の分岐予測を行う分岐予測機構と、命令をデコードする命令デコード機構とを備え、命令のデコードと、条件分岐命令に対して上記分岐予測機構の出力に従い第1の分岐処理を行うか上記第1の分岐処理を行わないかかどちらかの動作とを行う機構をもつ第1のパイプラインステージと、条件分岐命令に対する分岐条件に従い、第2の分岐処理を行うかまたは上記第2の分岐処理を行わないかどちらかの処理を行う第2のパイプラインステージと、上記第1の分岐処理動作を許可または禁止することを制御する分岐制御情報を記憶するレジスタとを備え、上記分岐制御情報を書き換える命令をもつことを特徴とするデータ処理装置。

(2) 条件分岐命令の分岐履歴を保持する分岐履歴テーブルと、上記分岐履歴テーブルの出力値をも

とに分岐予測を行う分岐予測機構と、命令をデコードする命令デコード機構とを備え、命令のデコードと、条件分岐命令に対して上記分岐予測機構の出力に従い第1の分岐処理を行うか上記第1の分岐処理を行わないかかどちらかの動作とを行う機構をもつ第1のパイプラインステージと、条件分岐命令に対する分岐条件に従い、第2の分岐処理を行うかまたは上記第2の分岐処理を行わないかどちらかの処理を行う第2のパイプラインステージと、上記分岐履歴を書き換える命令と、を備えることを特徴とするデータ処理装置。

(3) オペランドバイパス制御情報を保持する制御レジスタと、フェッチする第1のオペランドの第1のアドレス値とストアする第2のオペランドの第2のアドレス値とを比較するアドレス比較機構と、上記アドレス比較機構での比較結果と上記制御レジスタの出力とを入力としてその論理演算結果を出力する論理回路と、上記論理演算結果が「バイパス可能」を示すとき、メモリをバイパスして上記第2のオペランドの値を上記第1のオペラン

## 特開平1-216427(2)

ドとするバイパス機構と、を備え、上記オペランドバイパス制御情報を書き換える命令をもつことを特徴とするデータ処理装置。

(4) フェッチすべきオペランドのアドレスを保持するアドレスレジスタと、ストアするオペランドのアドレスを送る第1のバスと、ストアするオペランドとフェッチするオペランドの両方を送る第2のバスと、第1のバスに接続し、第1のバスの値を書き込むことが可能な第1記憶装置と、第2のバスに接続し、第2のバスの値を書き込むことが可能な第2記憶装置と、上記アドレスレジスタに保持された第1のアドレス値と上記第1のバスで送られるストアするオペランドの第2のアドレス値とを比較する比較装置とを備え、オペランドをストアするとき、上記第1のアドレス値と第2のアドレス値が一致する場合には、上記第2のアドレス値を上記第1の記憶装置に書き込み、上記第2のバスで送られるオペランドを上記第2の記憶装置に書き込むことを特徴とするデータ処理装置。

Aステージ02は命令コード中で指定されたオペランドの実効アドレスの計算を行い、計算したオペランドアドレスをFステージ04に出力する。Fステージ04はAステージ03から入力されたオペランドアドレスに従い、メモリよりオペランドをフェッチする。フェッチしたオペランドはEステージ05に出力する。Eステージ05はFステージ04から入力されたオペランドに対して命令コード中で指定された演算を実行する。さらに必要ならその演算結果をメモリにストアする。

上記のパイプライン処理機構により、各命令で指定される処理は5つに分解され、5つの処理を順番に実行することにより、指定された処理を完了する。各々5つの処理は異なる命令に対して並列動作させることが可能であり、理想的には上記の5段のパイプライン処理機構により5つの命令を同時に処理し、パイプライン処理を行わない場合に比べ、最大で5倍の処理能力をもつデータ処理装置を得ることができる。

パイプライン処理技術は上記のようにデータ処

## 3. 発明の詳細な説明

## 〔産業上の利用分野〕

この発明はパイプラインの乱れを少なくする分岐命令処理機構および、オペランドのフェッチ動作を効率的に行うためのオペランド・バイパス機構により、多段パイプライン処理機構を効率的に動作させ、高い処理能力を実現したデータ処理装置に関するものである。

## 〔従来の技術〕

図7図に従来のデータ処理装置で行われていたパイプライン処理機構の例を示す。01は命令フェッチステージ(IPステージ)、02は命令デコードステージ(Dステージ)、03はオペランドアドレス計算ステージ(Aステージ)、04はオペランドフェッチステージ(Fステージ)、05は命令実行ステージ(Eステージ)である。

IPステージ01はメモリから命令コードをフェッチしてDステージ02に出力する。Dステージ02はIPステージ01から入力される命令コードをデコードして、デコード結果をAステージ03に出力する。

処理装置の処理能力を大幅に向上させる可能性をもつものであり、高速なデータ処理装置で広く用いられている。

しかし、パイプライン処理にもいくつかの欠点があり、いつも理想的な状態で命令が処理されるわけではない。パイプライン処理で問題となるものの1つは命令のシーケンスを乱す分岐命令の実行である。

図7図に示すパイプライン処理機構をもち、分岐命令をEステージ05で処理してから分岐命令をIPステージ01が処理する従来のデータ処理装置では、分岐命令の実行により、パイプラインが大幅に乱れる。従来のデータ処理装置で分岐命令が実行された場合に、パイプライン中を流れる命令の様子を図8図に示す。図8図では命令3及び命令12が分岐命令である。命令3が実行されるとすでにパイプライン処理中の命令4、命令5、命令7はキャンセルされ、新たに命令11がIPステージ01から処理される。命令3がEステージ05で実行されてから命令11がEステージ05で実行されるまで

## 特開平 1-216427 (3)

には 4 命令処理分の時間が無駄になる。命令 12 についても同様に 4 命令処理分の時間が無駄になる。この無駄時間は分岐命令の実行後に処理すべき命令のフェッチが分岐命令に対する全パイプライン処理が終了した後に行われるためであり、パイプライン処理の段数が多いほどこの無駄時間も長くなる。

このような、分岐命令によるパイプラインの乱れを少なくするために、D ステージ 0 において命令をデコードする段階で条件分岐命令の分岐を予測し、その条件分岐命令が E ステージ 0 で実行される前に IF ステージ 0 による命令のフェッチ先をあらかじめ分岐先に変更する分岐予測処理が従来より行われている。

また、F ステージ 0 でプリフェッチしようとしているオペランドと E ステージ 0 が書き込み処理を行おうとするオペランドが一致するとき、F ステージ 0 によるオペランドのプリフェッチはメモリから行わず、E ステージが書き込もうとしているオペランド値をバイパスする操作をおこなう。

るため、命令の流れを追うのが困難になる。

同様に、オペランド・フェッチを外部アドレス・バスでモニタする場合、オペランドのバイパスが起こると、F ステージのメモリ・アクセスが外部には見えなくなってしまう。

〔課題を解決するための手段〕

前記問題点を解決するために、本発明においては、分岐予測機構を動作させるかどうかの指定ができるようにし、また、分岐履歴を初期化できるようにした。

アドレス比較機構とバスサイクルごとのアドレスとデータを格納する機構により、効率的にバイパスが行えるオペランド・バイパス機構を設け、このオペランド・バイパス機構を動作させるかどうかの指定ができるようにした。

さらに、これらの指定をソフトウェアにより専用のレジスタの特定のビットに値を設定することで行えるようにした。

また、ある特定の命令を実行した場合に、暗黙のうちに分岐履歴をクリアする機能も設けた。

この操作によってメモリをリードする動作が 1 回省ける。

〔発明が解決しようとする課題〕

分岐予測を行うことにより、条件分岐命令の分岐を予測し、命令フェッチをあらかじめ分岐先に切り替えることで、パイプラインの処理効率は一時的に上がる。

しかし、条件分岐命令で分岐する場合と分岐しない場合が交互に起こるようなアプリケーション・プログラムなどでは、分岐予測をすることによってかえって効率が悪くなることもある。

システムのリセット時やプログラムの走る状況が変化した場合などは、分岐予測ビットが以前に走っていたプログラムの履歴に従ったものになっているので全くあてにならない。

また、マイクロプロセッサの動作をチップの外部ピンでモニタし、デバッグするような場合、外部アドレス・バスをモニタすると、分岐予測機構が分岐すると予測して外れた場合、1 度分岐が起こり、再び分岐してもとの流れに戻るよう見え

〔作用〕

本発明における手段を用いることによって、分岐予測機構を用いることでかえって処理効率が落ちる場合や、外部アドレス・バスをモニタする時には、制御レジスタへ値をセットすることにより、分岐予測機構を動作させないようにすることができ

る。プログラムの走る状況が変化するなどして、分岐履歴があてにならない場合は、制御レジスタへ値を書き込むことで分岐履歴をクリアできる。また、プログラムの実行状況を変化させるような命令（コンテキスト・スイッチ用の命令など）が実行された場合には、分岐履歴が自動的にクリアされる。

オペランド・フェッチを外部アドレス・バスでモニタする場合などは、制御レジスタへ値をセットすることにより、オペランド・バイパス動作をさせないようにすることができる。

〔発明の実施例〕

(1) 実施例 1 の構成

## 特開平1-216427 (4)

第5図に本発明のデータ処理装置のブロック図を示す。本発明処理装置の内部を機能的に大きく分けると、命令フェッチ部50、命令デコード部52、PC計算部54、オペランドアドレス計算部56、マイクロROM部58、データ演算部60、外部バスインターフェイス部62に分かれる。第5図では、その他にCPU外部にアドレスを出力するアドレス出力回路64とCPU外部とデータの入出力を行うデータ入出力回路66を他の機能ブロック部と分けて示した。

## (1.1) 命令フェッチ部

命令フェッチ部50にはブランチバッファ、命令キューとその制御部などがあり、次にフェッチすべき命令のアドレスを決定して、ブランチバッファやCPU外部のメモリから命令をフェッチする。ブランチバッファへの命令登録も行う。

ブランチバッファは小規模であるためセレクトイブキャッシュとして動作する。ブランチバッファの動作の詳細は特願昭61-202041で詳しく述べられている。

次にフェッチすべき命令のアドレスは命令キュー

さらにPHWデコーダやNPHWデコーダの出力をさらにデコードして、マイクロROMのエントリアドレスを計算するデコーダ2、条件分岐命令の分岐予測を行う分岐予測機構、オペランドアドレス計算のときのパイプラインコンフリクトをチェックするアドレス計算コンフリクトチェック機構も含まれる。

命令フェッチ部より入力された命令コードを2クロックにつき0～6バイトのデコードする。デコード結果のうち、データ演算部60での演算に関する情報がマイクロROM部58に、オペランドアドレス計算に関する情報がオペランドアドレス計算部56に、PC計算に関する情報がPC計算部54に、それぞれ出力される。

## (1.3) マイクロROM部

マイクロROM部58には主にデータ演算部60を制御するマイクロプログラムが格納されているマイクロROM、マイクロシーケンサ、マイクロ命令デコーダなどが含まれる。マイクロ命令はマイクロROMから2クロックに1度読み出される。マイク

ーに入力すべき命令のアドレスとして専用のカウンタで計算される。分岐ジャンプが起きたときには、新たな命令のアドレスが、PC計算部54やデータ演算部60より伝送されてくる。

CPU外部のメモリから命令をフェッチするときには、外部バスインターフェイス部62を通して、フェッチすべき命令のアドレスをアドレス出力回路64からCPU外部に出力し、データ入出力回路66から命令コードをフェッチする。

バッファリングした命令コードのうち、命令デコード部52で次にデコードすべき命令コードを命令デコード部52に出力する。

## (1.2) 命令デコード部

命令デコード部52では基本的に16ビット(ハーフワード)単位に命令コードをデコードする。このブロックには第1ハーフワードに含まれるオペコードをデコードするPHWデコーダ、第2、第3ハーフワードに含まれるオペコードをデコードするNPHWデコーダ、アドレッシングモードをデコードするアドレッシングモードデコーダが含まれる。

ロシーケンサはマイクロプログラムで示されるシーケンス処理の他に、例外、割込、トラップ(この3つをあわせてEITと呼ぶ)の処理をハードウェア的に受付ける。またマイクロROM部58はストアバッファの管理も行う。マイクロROM部58には命令コードに依存しない割込みや演算実行結果によるフラッグ情報と、デコーダ2の出力など命令デコード部の出力が入力される。マイクロデコーダの出力は主にデータ演算部60に対して出力されるが、ジャンプ命令の実行による他の先行処理中止情報など一部の情報は他のブロックへも出力される。

## (1.4) オペランドアドレス計算部

オペランドアドレス計算部56は、命令デコード部52のアドレッシングモードデコーダなどから出力されたオペランドアドレス計算に関する情報によりハードワイヤード制御される。このブロックではオペランドのアドレス計算に関するほとんどの処理が行われる。メモリ間接アドレッシングのためのメモリアクセスのアドレスやオペランドアドレスがメモリにマップされたI/O領域に入るか

## 特開平 1-216427 (5)

どうかのチェックも行われる。

アドレス計算結果は外部バスインターフェイス部50に送られる。アドレス計算に必要な汎用レジスタやプログラムカウンタの値はデータ演算部より入力される。

メモリ間接アドレッシングを行うときは外部バスインターフェイス部50を通してアドレス出力回路51からCPU外部に参照すべきメモリアドレスを出力し、データ入出力回路52から入力された間接アドレス値を命令デコード部53をそのまま通過させてフェッチする。

## (1.5) PC計算部

PC計算部54は命令デコード部53から出力されるPC計算に関する情報でハードワイヤードに制御され、命令のPC値を計算する。本特許のデータ処理装置は可変長命令セットを持っており、命令をデコードしてみないとその命令の長さが判らない。PC計算部54は、命令デコード部53から出力される命令長をデコード中の命令のPC値に加算することによりつぎの命令のPC値を作り出す。また、命令

演算部55で実行する。演算対象となるオペランドがアドレスや即値の場合は、オペランドアドレス計算部56で計算されたアドレスや即値を外部バスインターフェイス部50を通過させて得る。また、演算対象となるオペランドがCPU外部のメモリにあるデータの場合は、アドレス計算部56で計算されたアドレスをバスインターフェイス部50がアドレス出力回路51から出力して、CPU外部のメモリからフェッチしたオペランドをデータ入出力回路52から得る。

演算部55としてはALU、バレルシフタ、プライオリティエンコーダやカウンタ、シフトレジスタなどがある。レジスタと主な演算部55の間は3バスで結合されており、1つのレジスタ間演算を指示する1マイクロ命令を2クロックサイクルで処理する。

データ演算のときCPU外部のメモリをアクセスする必要がある時はマイクロプログラムの指示により外部バスインターフェイス部50を通して、アドレス出力回路51からアドレスをCPU外部に出力し、

デコード部53が、分岐命令をデコードしてデコード回路57での分岐を指示したときは命令長の代わりに分岐変位を分岐命令のPC値に加算することにより分岐先命令のPC値を計算する。分岐命令に対して命令デコード段階で分岐を行うことを本発明のデータ処理装置ではプリブランチと呼ぶ。プリブランチの方法については特願61-204500と特願61-200557で詳しく述べられている。

PC計算部54の計算結果は各命令のPC値として命令のデコード結果とともに出力されるほか、プリブランチ時には、次にデコードすべき命令のアドレスとして命令フェッチ部58に出力される。

また、次に命令デコード部53でデコードされる命令の分岐予測のためのアドレスにも使用される。分岐予測の方法については特願62-8394で詳しく述べられている。

## (1.6) データ演算部

データ演算部55はマイクロプログラムにより制御され、マイクロROM部59の出力情報に従い、各命令の機能を実現するために必要な演算をレジスタと

データ入出力回路52を通して目的のデータをフェッチする。

CPU外部のメモリからデータをリードするときは、アドレスをAA1レジスタ60に設定し、そのアドレスを外部バスインターフェイス部50を通してアドレス入出力回路51より出力し、データをデータ入出力回路52からDDバス(108)を通してDDR1レジスタに取り込む。

CPU外部のメモリへデータをライトするときは、アドレスをAA1レジスタ60に設定し、そのアドレスを外部バスインターフェイス部50を通してアドレス入出力回路51より出力し、DDR2レジスタ61に設定したデータをDDバス(108)を通してデータ入出力回路52よりCPU外部に出力する。

ジャンプ命令の処理や例外処理などを行って新たな命令アドレスをデータ演算部55が得たときはこれを命令フェッチ部58とPC計算部54に出力する。

## (1.7) 外部バスインターフェイス部

外部バスインターフェイス部50は本特許のデータ処理装置の外部バスでの通信を制御する。メモ

## 特開平1-216427 (6)

りのアクセスはすべてクロック同期で行われ、最小2クロックサイクルで行うことができる。

メモリに対するアクセス要求は命令フェッチ部60、アドレス計算部64、データ演算部66から独立に生じる。さらに、オペランドプリフェッチを行うためのアクセス要求も生じる。外部バスインターフェイス部62はこれらのメモリアクセス要求を調停する。さらにメモリとCPUを結ぶデータバスサイズである32ビット(ワード)の境界をまたぐメモリ番地にあるデータのアクセスは、このブロック内で自動的にワード境界をまたぐことを検知して、2回のメモリアクセスに分解して行う。

プリフェッチするオペランドとストアするオペランドが異なる場合の、コンフリクト防止処理やストアオペランドからフェッチオペランドへのパイプライン処理も行う。

命令フェッチ部60からのアクセス要求がある場合は、CAAレジスタ(142)にアドレスが設定される。アドレス計算部64からのアクセス要求がある場合は、IAレジスタ(126)にアドレスが設定される。デ

ータ演算部66からのアクセス要求がある場合は、AA1レジスタ70にアドレスが設定される。

オペランドのプリフェッチを行うためのアクセス要求がある場合は、PAレジスタ(127)に設定されたアドレスがAAバス(107)に出力され、CPU外部のメモリからオペランドデータがフェッチされる。フェッチされたオペランドデータはDDバス(108)を流してSDATA40に入力される。また、アクセスに用いたAAバス(107)上のアドレスがSCAM40に入力される。SCAM40とSDATA40は一次指示線40でつながっている。SDATA40には逐地された4バイトのデータが2つまで入る。SCAM40には、SDATA40中のデータに対応するアドレスが入る。SDATA40へのデータの入力は逐地されてなされるが、そのデータをデータ演算部66が取り出して使う時には、任意のアドレスから任意のデータ長(ただし4バイト以内)で取り出しが行える。

## 図パイプライン機構

本発明のデータ処理装置のパイプライン処理は図6図に示す構成となる。命令のプリフェッチを

行う命令フェッチステージ(IFステージ60)、命令のデコードを行うデコードステージ(Dステージ62)、オペランドのアドレス計算を行うオペランドアドレス計算ステージ(Aステージ64)、マイクロROMアクセス(特にRステージ66と呼ぶ)とオペランドのプリフェッチ(特にOPステージ68と呼ぶ)を行うオペランドフェッチステージ(Fステージ64)、命令の実行を行う実行ステージ(Eステージ66)の5段階構成をパイプライン処理の基本とする。Eステージ66では1段のストアバッファがあるほか、高機能命令の一部は命令実行自体をパイプライン化するため、実際には5段以上のパイプライン処理効果がある。

各ステージは他のステージとは独立に動作し、理論上は5つのステージが完全に独立動作する。各ステージは1回の処理を最小2クロックで行うことができる。従って理想的には2クロックごとに次々とパイプライン処理が進行する。

本発明のデータ処理装置にはメモリーメモリー間演算や、メモリー間接アドレッシングなど、基本パ

イプライン処理1回だけでは処理が行えない命令があるが、本発明のデータ処理装置はこれらの処理に対してもなるべく均衡したパイプライン処理が行えるように設計されている。複数のメモリーオペランドをもつ命令に対してはメモリーオペランドの数をもとに、デコード段階で複数のパイプライン処理単位(ステップコード)に分解してパイプライン処理を行うのである。パイプライン処理単位の分解方法に関しては特願861-236456で詳しく述べられている。

IFステージ60からDステージ62に渡される情報は命令コードそのものである。Dステージ62からAステージ64に渡される情報は命令で指定された演算に関するもの(Dコード62と呼ぶ)と、オペランドのアドレス計算に関係するもの(Aコード64と呼ぶ)との2つある。Aステージ64からFステージ64に渡される情報はマイクロプログラムルーチンのエントリ番地やマイクロプログラムへのパラメータなどを含むRコード64と、オペランドのアドレスとアクセス方法指示情報などを含むFコード64



## 特開平1-216427 (7)

との2つである。Fステージ04からEステージ03に渡される情報は演算制御情報とリテラルなどを含むEコード04と、オペランドやオペランドアドレスなどを含むSコード04との2つである。

Eステージ03以外のステージで検出されたEITはそのコードがEステージ03の到達するまではEIT処理を起動しない。Eステージ03で処理されている命令のみが実行段階の命令であり、IFステージ01～Fステージ04で処理されている命令はまだ実行段階に到っていないのである。従ってEステージ03以外で検出されたEITは検出したことをステップコード中に記録して次のステージに伝えられるのみである。

## (2.1) バイプライン処理単位

第6図は、本発明におけるバイプライン処理ステージを示す。図において、01はIFステージ、02はDステージ、03はAステージ、04はFステージ、05はEステージ、06はRステージ、07はOFステージ、である。

## (2.1.1) ステップコードへの命令の分解

ードに付属してバイプラインの各ステージを流れるプログラムカウンタ値はステッププログラムカウンタ (SPC) と呼ぶ。SPCはバイプラインステージを次々と受け渡されていく。

## (2.2) 各バイプラインステージの処理

各バイプラインステージの入出力ステップコードには第6図に示したように便宜上名前が付けられている。またステップコードはオペコードに関する処理を行い、マイクロROMのエントリ番地やEステージ03に対するパラメータなどになる系列とEステージ03のマイクロ命令に対するオペランドになる系列の2系列がある。

## (2.2.1) 命令フェッチステージ

命令フェッチステージ (IFステージ01) は命令をメモリやブランチバッファからフェッチし、命令キューに入力して、Dステージ02に対して命令コードを出力する。命令キューの入力は4バイト単位で行う。メモリから命令をフェッチするときには4バイトにつき最小2クロックを要する。ブランチバッファがヒットした

本発明のデータ処理装置では上記の命令フォーマットの特徴を生かしたバイプライン処理を行う。Dステージ02では (2バイトの命令基本部+0~4バイトのアドレッシング修飾部)、(多段間接モード指定部+アドレッシング修飾部) または命令固有の修飾部を1つのデコード単位として処理する。各回のデコード結果をステップコードと呼び、Aステージ03以降ではこのステップコードをバイプライン処理の単位としている。ステップコードの数は命令ごとに関有であり、多段間接モード指定を行わないとき、1つの命令は最小1個、最大3個のステップコードに分かれる。多段間接モード指定があればそれだけステップコードが増える。

## (2.1.2) プログラムカウンタの管理

本発明のデータ処理装置のバイプライン上に存在するステップコードはすべて別命令に対するものである可能性があり、プログラムカウンタの値はステップコードごとに管理する。すべてのステップコードはそのステップコードのもとになった命令のプログラムカウンタ値をもつ。ステップコ

時は4バイトにつき1クロックでフェッチ可能である。命令キューの出力単位は2バイトごとに可変であり、2クロックの間に最大6バイトまで出力できる。また分岐の直後には命令キューをバイパスして命令基本部2バイトを直接命令デコードに転送することもできる。

ブランチバッファへの命令の登録やクリアなどの制御、プリフェッチ先命令アドレスの管理や命令キューの制御もIFステージ01で行う。

IFステージ01で検出するEITには命令をメモリからフェッチするときのバスアクセス例外や、メモリ保護違反などによるアドレス変換例外がある。

## (2.2.2) 命令デコードステージ

命令デコードステージ (Dステージ02) はIFステージ01から入力された命令コードをデコードする。デコードは命令デコード部02のPHWデコード、NPHWデコード、アドレッシングモードデコードを使用して、2クロック単位に1度行ない、1回のデコード処理で、0~6バイトの命令コードを消費する (RET命令の復帰先アドレスを含むステップ

## 特開平1-216427 (8)

コードの出力処理などでは命令コードを消滅しない)。1回のデコードでAステージ毎に対してアドレス計算情報であるAコード毎である約35ビットの制御コードと最大32ビットアドレス修飾情報と、オペコードの中間デコード結果であるDコード毎である約50ビットの制御コードと8ビットのリテラル情報と、を出力する。

Dステージ毎では各命令のPC計算部毎の制御、分岐予測処理、プリブランチ命令に対するプリブランチ処理、命令キューからの命令コード出力処理も行う。

Dステージ毎で検出するEITには予約命令例外、プリブランチ時の奇数アドレスジャンプトラップがある。また、IFステージOPより伝送されてきた各種EITはステップコード内にエンコードする処理をしてAステージ毎に伝送する。

## (2.2.3) オペランドアドレス計算ステージ

オペランドアドレス計算ステージ(Aステージ毎)は処理が大きく2つに分かれる。1つは命令デコード部毎のデコード2を使用して、オペコー

して出力する。この際、アドレス計算に伴うレジスタやメモリの読み出し時にコンフリクトチェックを行い、先行命令がレジスタやメモリに書き込み処理を終了していないためコンフリクトが指示されれば、先行命令がEステージ毎で書き込み処理を終了するまで待つ。また、オペランドアドレスやメモリ間接参照のアドレスがメモリにマップされたI/O領域に入るかどうかのチェックも行う。

Aステージ毎で検出するEITには予約命令例外、特権命令例外、バスアクセス例外、アドレス変換例外、メモリ間接アドレッシングのときのオペランドブレイクポイントヒットによるデバグトラップがある。Dコード毎、Aコード毎自体がEITを起こしたことを示しておれば、Aステージ毎はそのコードに対してアドレス計算処理をせず、そのEITをRコード毎やPコード毎に伝える。

## (2.2.4) マイクロROMアクセスステージ

オペランドフェッチステージ(Pステージ30も)処理が大きく2つに分かれる。1つはマイクロROMのアクセス処理であり、特にRステージ毎と呼ぶ。

Dの後段デコードを行う処理で、他方はオペランドアドレス計算部64でオペランドアドレスの計算を行う処理である。

オペコードの後段デコード処理はDコード毎を入力とし、レジスタやメモリの書き込み予約及びマイクロプログラムのエントリ落地とマイクロプログラムに対するパラメータなどを含むRコード毎の出力を行う。なお、レジスタやメモリの書き込み予約は、アドレス計算で参照したレジスタやメモリの内容が、パイプライン上を先行する命令で書き換えられ、誤ったアドレス計算が行われるのを防ぐためのものである。レジスタやメモリの書き込み予約はアドレスロックを避けるため、ステップコードごとに行うのではなく命令ごとに行う。レジスタやメモリの書き込み予約については特願62-144394で詳しく述べられている。

オペランドアドレス計算処理はAコード毎を入力とし、Aコード毎に従いオペランドアドレス計算部64で加算やメモリ間接参照を組み合わせてアドレス計算を行い、その計算結果をPコード毎と

他方はオペランドプリフェッチ処理であり、特にOPステージ毎と呼ぶ。Rステージ毎とOPステージ毎は必ずしも同時に動作するわけではなく、メモリアクセス権が獲得できるかどうかなどに依存して、独立に動作する。

Rステージ毎の処理であるマイクロROMアクセス処理はRコード毎に対して次のEステージでの実行に使用する実行制御コードであるEコードを作り出すためのマイクロROMアクセスとマイクロ命令デコード処理である。1つのRコード毎に対する処理が2つ以上のマイクロプログラムステップに分解される場合、マイクロROMはEステージ毎で使用され、次のRコード毎はマイクロROMアクセス待ちになる。Rコード毎に対するマイクロROMアクセスが行われるのはその前のEステージ毎の最後のマイクロ命令実行の時である。本発明のデータ処理装置ではほとんどの基本命令は1マイクロプログラムステップ行われるため実際にはRコード毎に対するマイクロROMアクセスが次々と行われることがない。



## 特開平1-216427 (9)

Rステージで新たに検出するEITはない。Rコードが命令処理再実行型のEITを示しているときはそのEIT処理に対するマイクロプログラムが実行されるのでRステージはそのRコードに従ったマイクロ命令をフェッチする。Rコードが奇数アドレスジャンプトラップを示しているときRステージはそれをEコードに伝える。これはブリランチに対するもので、EステージではそのEコードで分岐が生じなければそのブリランチを有効として奇数アドレスジャンプトラップを発生する。

## (2.2.5) オペランドフェッチステージ

オペランドフェッチステージ(OPステージ)はPステージで行う上記の2つの処理のうちオペランドプリフェッチ処理を行う。

オペランドプリフェッチはPコードを入力とし、フェッチしたオペランドとそのアドレスをSコードとして出力する。1つのPコードではワード境界をまたいでもよいが4バイト以下のオペランドフェッチを指定する。Pコードにはオ

ペランドのアクセスを行うかどうかの指定も含まれており、Aステージで計算したオペランドアドレス自体や即値をEステージに転送する場合にはオペランドプリフェッチは行わず、Pコードの内容をSコードとして転送する。プリフェッチしようとするオペランドとEステージが書き込み処理を行おうとするオペランドが一致するときは、オペランドプリフェッチはメモリから行わず、バイパスして行なう。またI/O領域に対してはオペランドプリフェッチを遮断させ、先行命令がすべて完了するまで待ってオペランドフェッチを行う。

OPステージで検出するEITにはバスアクセス例外、アドレス変換例外、オペランドプリフェッチに対するブレイクポイントヒットによるデバッグトラップがある。Pコードがデバッグトラップ以外のEITを示しているときはそれをSコードに転送し、オペランドプリフェッチは行わない。Pコードがデバッグトラップを示しているときはそのPコードに対してEITを示していないと

と同じ処理をすると共にデバッグトラップをSコードに伝える。

## (2.2.6) 実行ステージ

実行ステージ(Eステージ)はEコード、Sコードを入力として動作する。このEステージが命令を実行するステージであり、Pステージ以前のステージで行われた処理はすべてEステージのための前処理である。Eステージでジャンプ命令が実行されたり、EIT処理が起動されたりしたときは、IPステージGP~Pステージまでの処理はすべて無効化される。Eステージはマイクロプログラムにより制御され、Rコードに示されたマイクロプログラムのエントリ番地からの一連のマイクロプログラムを実行することにより命令を実行する。

マイクロROMの読み出しとマイクロ命令の実行はパイプライン化されて行われる。従ってマイクロプログラムで分岐が起きたときは1マイクロステップの遅延ができる。また、Eステージはデータ格納部にあるストアバッファを利用して、

4バイト以内のオペランドストアと次のマイクロ命令実行をパイプライン処理することもある。

EステージではAステージで行ったレジスタやメモリに対する書き込み予約をオペランドの書き込みの後、解除する。

また条件分岐命令がEステージで分岐を起こしたときはその条件分岐命令に対する分岐予測が誤っていたのであるから分岐履歴の書き換えを行う。

Eステージで検出されるEITにはバスアクセス例外、アドレス変換例外、デバッグトラップ、奇数アドレスジャンプトラップ、予約機能例外、不正オペランド例外、予約スタックフォーマット例外、ゼロ除算トラップ、無条件トラップ、条件トラップ、遅延コンテキストトラップ、外部割込、遅延割込、リセット割込、システム障害がある。

Eステージで検出されたEITはすべてEIT処理されるがEステージ以前のIPステージGP~Pステージの間で検出されRコードやSコードに反映されているEITは必ずEIT処理されるとは限らない。IPステージGP~Pステージの間で検

## 特開平1-216427 (10)

出したが、先行の命令がEステージ内でジャンプ命令が実行されたなどの原因でEステージ内で到達しなかったBITはすべてキャンセルされる。そのBITを起こした命令はそもそも実行されなかったことになる。

外部用込や遅延割込は命令の切れ目でEステージ内で直接受け付けられ、マイクロプログラムにより必要な処理が実行される。その他の各種BITも処理はマイクロプログラムにより行われる。

## (23) 各パイプラインステージの状態制御

パイプラインの各ステージは入力ラッチと出力ラッチを持ち、他のステージとは独立に動作することを基本とする。各ステージは1つ前に行った処理が終わり、その処理結果を出力ラッチから次のステージの入力ラッチに伝送し、自分のステージの入力ラッチに次の処理に必要な入力信号がすべてそろえば次の処理を開始する。

つまり、各ステージは、1つ前段のステージから出力されてくる次の処理に対する入力信号がすべて有効となり、今の処理結果を後段のステージ

へ処理を行う。動的な分岐予測処理は実行ステージで分岐を行う代わりにデコードステージで分岐を行うことによりなるべく早く分岐先命令を取り込むことを狙いとしている。

本発明のデータ処理装置に限らず、データ処理装置では、一般に分岐命令が実行される頻度は大きく、動的な分岐予測処理による性能改善効果は大きい。

## (3.1) 分岐命令の種類

本発明のデータ処理装置では動的な分岐予測処理を行う命令をプリランチ命令と呼ぶ。プリランチ命令には無条件分岐命令のように、動的な予測にかかわらず、必ず分岐する命令も含む。

本発明のデータ処理装置がもつ分岐命令は分岐条件がスタティックかダイナミックか及び分岐先がスタティックかダイナミックかにより計4種類に分類できるが、本発明のデータ処理装置ではこのうち2つの2種類に分類される命令をプリランチ命令とする。

第1の種類の分岐命令は分岐条件、分岐先とも

の入力ラッチに伝送して出力ラッチが空になると次の処理を開始する。

各ステージが動作を開始する1つ前のクロックタイミングで入力信号がすべてそろっている必要がある。入力信号がそろっていないと、そのステージは待ち状態(入力待ち)になる。出力ラッチから次のステージの入力ラッチへの伝送を行うときは次のステージの入力ラッチが空き状態になっている必要があり、次のステージの入力ラッチが空きでない場合もパイプラインステージは待ち状態(出力待ち)になる。必要なメモリアクセス値が獲得できなかったり、処理しているメモリアクセスにウェイトが挿入されたり、その他のパイプラインコンフリクトが生じると、各ステージの処理自体が遅延する。

## (3) 分岐命令の処理

本発明のデータ処理装置では上記に説明したように多段のパイプライン処理を採用しているため分岐命令を実行した際のオーバーヘッドが大きい。このオーバーヘッドを小さくするため動的な分岐予

スタティックな命令である。この種の命令には無条件分岐命令(BRA)とサブルーチン呼び出し命令(BSR)がある。第2の種類の分岐命令は分岐条件がダイナミックで分岐先がスタティックな命令である。この種の命令には条件分岐命令(Bcc)、ループ制御命令(ACB)がある。

## (3.2) 分岐命令処理回路の機能構成

第4図に本発明のデータ処理装置の分岐命令処理回路の構成を示す。第4図には命令フェッチ部50、命令デコード部52、PC計算部54、オペランドアドレス計算部56、データ演算部58、外部バスインターフェイス部59のそれぞれに含まれる回路の部分詳細図と、アドレス出力回路60、データ入力回路61よりなる。

命令デコーダ(111)とPC加算器(132)の入力側、アドレス加算器(124)の入力側は、ディスプレースメント値、分岐命令の定位置を伝送するDISPバス60で結ばれている。命令デコーダ(111)とアドレス加算器(124)の入力側はステップコード生成に使用した命令コード長、スタックプッシュモードのと

## 特開平1-216427 (11)

きのプリデクリメント値などを伝送する補正値バス(101)でも結ばれている。命令デコーダ(111)とPC加算器(132)の入力側はステップコード生成に使用した命令コード長を伝送する命令長バス(101)でも結ばれている。レジスタファイル(144)とアドレス加算器(124)入力側はレジスタファイル(144)に格えられているアドレス値を伝送するAバス(103)で結ばれている。

命令デコーダ(111)には命令キュー(112)から命令コードが入力され、分岐予測テーブル(113)から分岐予測ビットが入力される。命令デコーダ(111)の出力部には、分岐予測結果により、条件分岐命令の分岐条件指定フィールドを、Eステージ側でそのまま出力するか条件指定を反転して出力するかを選択を行う、分岐条件生成回路(114)がある。

命令長バス(101)の値とDISPバス側の値のどちらかを選択して入力する或加算値選択回路(131)の出力と、Dステージ側でデコードした命令のPC値を保持するDPC(135)またはステップコードの切れ目毎の作業用PC値を保持するTPC(134)のどちらかと、

力として、3値加算を行う。アドレス加算器(124)の出力値はアドレス加算器出力ラッチ(125)を通して、AOバス(106)に出力される。AOバス(106)は、メモリ間接アドレッシングを行うとき、AAバス(107)を通してアドレス出力回路側からCPU外部にアドレス値を出力するときそのアドレス値を保持するラッチIA(126)と、Fステージでのオペランドプリフェッチ時に、AAバス(107)を通してアドレス出力回路側からCPU外部にオペランドアドレス値を出力するとき、そのオペランドアドレス保持するラッチFA(127)と、につながる。

FA(127)は、アドレス加算器(124)で計算されたオペランドアドレスをEステージ側で使用するためにオペランドアドレス値を保持するラッチSA(141)への出力経路をもつ。SA(141)はデータ演算部側の汎用データバスであるSバス(109)への出力経路をもつ。命令のアドレスを伝送するCAバス(104)はPC加算器出力ラッチ(133)と、TPC(134)と、命令フェッチ回路がプリフェッチする命令コードの番地を管理するカウンタQINPC(115)と、命令フェッ

はPC加算器(132)に入力される。PC加算器(132)の出力はPC加算器出力ラッチ(133)を通してCAバス(104)やPOバス(105)に出力される。POバス(105)はラッチTPC(134)、ラッチDPC(135)、Aステージで処理中の命令のPC値を保持するラッチAPC(136)、さらに分岐予測テーブル(113)にも結合している。TPC(134)にはEステージ側で分岐やジャンプが生じたとき、新たな命令番地を入力するため、CAバス(103)からの入力経路もある。

補正値バス(102)の出力とDISPバス(100)の出力はディスプレイメント選択回路(122)に入力され、どちらか一方がアドレス加算器(124)に入力される。DISPバス側出力とAバス(103)出力はベースアドレス選択回路(123)に入力され、どちらか一方がアドレス加算器(124)に入力される。アドレス加算器(124)は、ディスプレイメント選択回路(122)の出力、ベースアドレス選択回路(123)の出力、それにAバス(103)より入力された値をシフトすることにより、1倍、2倍、4倍、8倍の値とするインデックス値生成回路(121)の出力、の計3つの値を入

ちのためのアドレスをAAバス(107)を通してアドレス出力回路側からCPU外部に出力するときその値を保持するラッチCAA(142)と、Eステージ側で分岐やジャンプが起きたときに新たな命令番地をSバス(109)から入力するラッチEB(143)と、に結合している。APC(136)はAバス(103)と、Fステージ側で処理中の命令のPC値を保持するラッチFPC(137)とに出力経路がある。FPC(137)はEステージ側で処理中の命令のPC値を保持するラッチCPC(138)への出力経路をもつ。CPC(138)はSバス(109)と、分岐履歴管理のためにPC値の最下位バイトの値を保持するラッチOPC(139)とに出力経路をもつ。レジスタファイル(144)は汎用レジスタや作業用レジスタなどからなり、Sバス(109)とAバス(103)への出力経路をもち、Dバス(110)から入力経路を持つ。データ演算部側の演算機構であるデータ演算器(145)はSバス(109)から入力経路をもち、Dバス(110)への出力経路を持つ。

## (3.3) 分岐予測方法

本発明のデータ処理装置では無条件分岐命令BRA、

サブルーチン分岐命令BSR、ループ制御命令ACB、3つの命令については、分岐予測テーブルの出力である分岐予測ビットにかかわらず、必ず分岐すると予測する。BRA、BSRに対してはこの予測は必ず正しい。

ACBはループ制御変数に指定された値を加えて、その結果がループ終了条件を満たすかどうかを判定し、ループ終了条件を満たさなければ分岐し、満たせば分岐しない命令である。従って、大多数のソフトウェアではACBについてもこの予測方法はかなりの確率で正しい。また、ACBに対する本発明のデータ処理装置の特徴的な処理を認識してソフトウェアを作成すれば認識しない場合より効率的なプログラムを作成することが可能である。

条件分岐命令Bccについては分岐するかしんないかを過去の履歴に従って判断する。履歴はBcc命令の1つ前に実行した命令のアドレスの下位8ビットのアドレスをもとに行う。分岐予測は過去1回の分岐履歴のみに従い、1ビットで示される。

(3.4) 分岐予測テーブルの構成

とになる。

本発明のデータ処理装置では、Dステージ④でデコードしようとする命令の1つ前にDステージ④でデコードされた命令のアドレスの下位8ビットをもとに分岐予測テーブル(113)を引いて分岐予測を行う。分岐予測は過去1回の履歴のみに従ったダイレクトマッピング方式で登録されている。本発明のデータ処理装置では命令アドレスの最下位ビット(右端のビット)は必ずゼロであるため分岐予測テーブルは128ビットで構成されている。

分岐予測ビットを有効に使用するのはBcc命令をデコードするときのみであるが、分岐予測ビットは、使用するかどうかにかかわらず、すべての命令の命令コードと共に命令デコード(111)に入力する。このため分岐予測テーブル(113)の参照は、デコードしようとする命令の1つ前の命令がデコードされているときPC加算器(132)から出力されてくる1つ前の命令のPC値の下位1バイト(最下位ビットは不要)で行う。これにより、分岐予測ビットは、次のDステージ処理の最初までに命令デ

## 特開平1-216427 (12)

第1図に分岐予測テーブル(113)の詳細を示す。POバス(105)からの入力7ビットとOPC(139)からの入力7ビットはセレクタ(151)を通して、デコーダ(152)に入力される。デコーダ(152)では7ビットを128ビットにデコードして128ビットの分岐履歴ラッチ(153)のうち1つを分岐予測値として分岐予測信号線(161)を通して分岐予測出力ラッチ(154)に出力する。128ビットの分岐履歴ラッチ(153)はクリア信号(157)が入力されると一斉に値をゼロにして「分岐しない」を示す。分岐予測出力ラッチ(154)は予測反転回路(155)によりその内容を反転して分岐予測更新ラッチ(156)に結合されている。分岐履歴ラッチ(153)には分岐予測動作制御信号線(160)が入力されている。この信号は、分岐履歴ラッチ(153)の書き込み許可信号となっており、この信号値が「L」の時は分岐履歴ラッチ(153)への書き込み動作を行わない。分岐予測信号線(161)は動作制御信号線(160)とANDをとって分岐予測出力ラッチ(154)に出力される。これによって、動作制御信号線(160)が「L」のときは常に「分岐しない」と予測されるこ

コード(111)に入力される。

分岐予測テーブル(113)の分岐履歴はクリア信号(157)により初期値をすべて「分岐しない」とできる。分岐予測の更新はBcc命令がEステージ④で分岐したときに行われる。Bcc命令がEステージ④で分岐を起こしたとき、それはDステージ④での分岐予測が間違っていたことを意味する。このときEステージ④で分岐予測の更新(間違っていた分岐履歴の反転)が行われる。Eステージ④ではOPC(139)の内容をデコーダ(152)に伝送し、そのデコード結果で対応する分岐履歴ラッチ(153)の内容を分岐予測出力ラッチ(154)に読み出す。次に、分岐予測出力ラッチ(154)の内容が反転された分岐予測更新ラッチ(156)の内容を、同じくOPC(139)の値で示された分岐履歴ラッチ(153)に書き換える。

分岐予測は対象となるBcc命令がデコードされる1つ前にデコードされた命令のPC値をもとに行われるため分岐予測テーブル(113)の更新もEステージ④でBcc命令の1つ前に実行された命令のPC値をもとに行う。このためEステージ④では現在

## 特開平1-216427 (13)

実行中の命令の1つ前に実行した命令のPC値の下位1バイト(最下位ビットは不変)を記憶しておくOPC(139)があり、分岐予測テーブル(113)の更新はこの値を用いて行う。分岐履歴の更新はEステージ6でBcc命令が分岐を起こしたときだけしか行われないため、Dステージ6の分岐予測テーブル(113)の参照動作がEステージ6の更新に妨げられることはない。Eステージ6で分岐が起きた直後はDステージ6はIFステージ6からの命令コード待ち状態となる。分岐履歴の書き換えは、この命令コード待ち状態の間に行われる。

分岐予測を行わない場合は、分岐予測動作制御信号線(160)の値を“L”にする。分岐予測信号線(161)は分岐予測動作制御信号線(160)とANDをとった結果、分岐予測出力ラッチ(154)に出力されるから、常に「分岐しない」と予測される。また、分岐予測動作制御信号線(160)は分岐履歴ラッチ(153)の書き込み制御信号となっているので、分岐履歴ラッチ(153)の書き込みも行われない。

## (3.5) PC計算部の動作

ス(101)に出力される。

プリブランチしない場合、Dステージ6は引き続き次の命令のデコード処理を行うと同時に、PC計算部6で引き続き次の命令のPC値を計算するため、TPC(134)の値と命令長バス(101)から伝送されたデコードで消費した命令コードの長さとの加算を行いTPC(134)に加算結果を書き換える。つまり、あるステップコードの先頭アドレスはそのステップコードがデコード処理により生成されたときに計算されるのである。プリブランチ以外ではデコードすべき命令コードは命令キュー(112)から次々と出力されるため、デコード開始段階でそのコードの先頭アドレスを知る必要はない。Dステージ6で生成したステップコードが命令Aの最後のステップコードであるとき、次の命令Bのデコード処理中に計算されるPC加算器(132)の出力は、命令Bの先頭番地であり、命令BのPC値であるから、PC加算器(132)の出力である命令BのPC値はPOバス(105)からTPC(134)とDPC(135)の両方に書き込まれる。さらにこのときAステージ6が入力コード待

PC計算部はDステージ6で命令コードがデコードされるとき、1つ前にデコードされた命令コードの長さ情報とその1つ前にデコードされた命令コードの先頭番地とからデコード中の命令コードの先頭番地を計算する。PC計算部ではDPC(135)に命令の切れ目のアドレスである命令のPC値を保持し、TPC(134)にステップコードの切れ目のアドレスを管理する。DPC(135)は命令の切れ目のアドレスが計算されたときだけ書き換えられる。TPC(134)はステップコードの切れ目のアドレス、つまり命令デコード処理ごとに書き換えられる。パイプライン上で処理されるステップコードのPC値はそのステップコードのもとになった命令のPC値が必要であるため、DPC(135)、FPC(138)と伝送されていく。

命令のデコードは(3.1.2)の項で述べたようにステップコード単位に行われ、1回のデコード処理で0~6バイトの命令コードが消費される。命令デコード処理ごとに判明したそのとき使用した命令コードの長さが命令デコーダ(111)から命令長バ

ス(101)に出力される。プリブランチしない場合、Dステージ6は引き続き次の命令のデコード処理を行うと同時に、PC計算部6で引き続き次の命令のPC値を計算するため、TPC(134)の値と命令長バス(101)から伝送されたデコードで消費した命令コードの長さとの加算を行いTPC(134)に加算結果を書き換える。つまり、あるステップコードの先頭アドレスはそのステップコードがデコード処理により生成されたときに計算されるのである。プリブランチ以外ではデコードすべき命令コードは命令キュー(112)から次々と出力されるため、デコード開始段階でそのコードの先頭アドレスを知る必要はない。Dステージ6で生成したステップコードが命令Aの最後のステップコードであるとき、次の命令Bのデコード処理中に計算されるPC加算器(132)の出力は、命令Bの先頭番地であり、命令BのPC値であるから、PC加算器(132)の出力である命令BのPC値はPOバス(105)からTPC(134)とDPC(135)の両方に書き込まれる。さらにこのときAステージ6が入力コード待

プリブランチする場合、Dステージ6はプリブランチ命令の最後のステップコードを出力した後、命令デコーダ(111)の処理を止め、分岐先命令のPC値を計算するため、DPC(135)の値とDISPバス6より伝送された分岐変位の加算を行う。さらに、IFステージ6に初期化指示を出し、加算結果である分岐命令のPC値をTPC(134)とDPC(135)に書き込むと共に、CAバス(104)にも出力してQINPC(115)、CAA(142)にも書き込む。

プリブランチによる分岐先命令アドレス計算の際、奇数アドレスジャンプトラップの検出も行ない、Dコード6中にその結果をパラメータとして示す。Eステージ6ではプリブランチが正しいと判明した時に、奇数アドレスジャンプトラップを起動する。プリブランチが間違っていて、再びEステージ6で分岐が生じたときはプリブランチで検出した奇数アドレスジャンプトラップは検出さ



## 特開平1-216427 (14)

れる。このため、Dステージ102で検出された奇数アドレスジャンプトラップはその他のEITとは別扱いとなっている。また、Eステージ104では奇数アドレスジャンプトラップの起動処理に奇数となった命令アドレスの値を必要とする。このため、Dステージ102は奇数アドレスジャンプトラップの検出を行ったとき、その奇数アドレス値をPC値とした特殊なステップコード(OAJTステップコード)を発生する。OAJTステップコードに対してAステージ106、Fステージ108はそのコードを次のステージに伝える。Eステージ104はプリブリランチが正しいと判断し、しかもそのプリブリランチが奇数アドレスジャンプトラップを検出しているとき、CPC(138)を通して次に伝送されてくるOAJTステップコードのPC値を使用して奇数アドレスジャンプトラップの起動処理を行う。

Eステージ104で分岐が生じたときは分岐先アドレスがEB(143)からCAバス(104)を介してTPC(134)に伝送されてくる。PC計算部100はこの値とゼロを加算して結果をPOバス(105)からTPC(134)、DPC(135)

に書き込む。これでPC計算部100の初期化が完了する。この初期化の処理はEステージ104で分岐が生じた最初の単位デコード処理とオーバーラップしてなされる。なお、QINPC(115)、CAA(142)にはCAバス(104)よりTPC(134)に値を取り込む際に同じ値がセットされる。

(3.7) プリブリランチ命令に対するオペランドアドレス計算部の動作

プリブリランチ命令に対して、Dステージ102がプリブリランチ処理を行わなかった場合、オペランドアドレス計算部104がプリブリランチ命令の分岐先アドレスを計算する。分岐先アドレスの計算は、Aバス(103)より伝送されてくるAPC(136)の値とDISPバス100より伝送されてくる分岐変位値をアドレス加算器(124)で加算することにより行われる。計算された分岐先アドレスはEステージ104に伝えられる。Aステージ106で、オペランドアドレス計算部104を使用した分岐先アドレスの計算の際は、奇数アドレスジャンプトラップの検出は行わない。Eステージ104に伝送される分岐先アドレスが奇数で

あることにより、奇数アドレスジャンプトラップの情報が伝えられるのである。

Dステージ102がプリブリランチ処理をした場合、Bcc命令、ACB命令に対しては、Aステージ106がそのプリブリランチ命令に引き続くアドレスにある次の命令のPC値を計算する。計算結果はEステージ104に伝えられ、プリブリランチが間違っていたときの再度の分岐先アドレスとして使用される。Bcc命令など、Dステージ102で1ステップコードにデコードされる命令に対しては、Aバス(104)より伝送されてくるAPC(136)の値に補正値バス(102)から伝送されてくるBcc命令の命令値を加算して、加算結果をAOバス(108)よりFA(127)に書き込む。

## 14) オペランド・バイパス機構

第2図は、外部バスインターフェイス制御部およびその周辺を示すブロック図である。図において、101はOFステージ制御部、103はEステージ制御部、105は外部バスインターフェイス制御部、107はOFステージ制御部からのアクセス要求信号、109はOFステージ制御部へのアクセス要求受付信号、111

はアクセス完了信号、113はEステージ制御部からのアクセス要求信号、115はEステージ制御部からのアクセス要求がリードかライトかを示す信号、117はEステージ制御部へのアクセス要求受付信号、119はオペランド・バイパス機構、121はオペランド・バイパス機構から出力されるバイパス可信号、123はオペランド・バイパス機構の動作を制御する信号、125はメモリアクセス制御回路である。

以下、第2図と第4図を用いてオペランドのバイパス処理を説明する。

外部バスインターフェイス制御部105はパイプラインの各ステージからバスアクセス要求を受け付けCPU外部のメモリとデータのやりとりを行う。

OFステージ制御部101がオペランドを外部メモリからプリフェッチする場合、OFステージ制御部101はFAレジスタ(127)にアドレスを設定して、外部バスインターフェイス制御部105に対してアクセス要求信号107を出す。外部バスインターフェイス制御部105は現在メモリ・アクセス中でなくかつ、Eステージ制御部103からのアクセス要求がない場合、OFス



## 特開平1-216427 (15)

ステージ制御部70へアクセス要求受付信号80を返す。メモリ・アクセスが完了すると外部バスインターフェース制御部70はOPステージ制御部70へアクセス完了信号80を返す。また、アクセスに使われたAAバス(107)の値はSCAM70に入力され、DDバス(108)上のデータはSDATA70に入力される。アクセスがワード境界をまたぐときは、アクセスは2回に渡って行われ、各アクセスのデータがそのままSDATA70に入力される。それと同時に各アクセスで使われたAAバス(107)のアドレスがそのままSCAM70に入力される。SDATA70に入力されたデータは、その後Sバスを通して演算部70へ取り込まれる。

Eステージ制御部70がデータをCPU外部のメモリから読み出す場合、Eステージ制御部70はAA1レジスタ70にアドレスを設定して、外部バスインターフェース制御部70に対してアクセス要求信号80およびリード/ライト信号80をリードにして出力する。外部バスインターフェース制御部70は、要求を受けるとEステージ制御部70へアクセス要求受付信号80を返す。メモリ・アクセスが完了

Eステージ制御部70がライト・アクセス要求を出し、同時にOPステージ制御部70がアクセス要求を出した場合、両方のアドレスが同じであるとする。Eステージ制御部70からの要求で書き込んだデータをその直後にOPステージ制御部70からの要求で読み出すことになり、外部バス・アクセスが1回分冗長になる。

そこで、AAバス(107)の値とPAレジスタ70の値を常に比較し、これらが一致しかつ、OPステージ制御部70からのアクセス要求とEステージ制御部70からのライト要求が同時に出力された場合、バイパス可信号80を出力するオペランド・バイパス機構70が設けられている。バイパス可信号80はOPステージ制御部70から見るとアクセス要求受付信号80と同じ意味を持つ。アクセス完了信号80によって、Eステージ制御部70とOPステージ制御部70両方からのアクセスが完了したことになり、AAバス(107)上のアドレスはSCAM70に入力され、DDバス(108)上のデータはSDATA70に入力される。

オペランド・バイパス機構70から出力される信

すると、外部バスインターフェース制御部70はEステージ制御部70へアクセス完了信号80を返す。また、DDバス(108)上のデータがDDR1レジスタ70に取り込まれる。

Eステージ制御部70がデータをCPU外部のメモリへ書き込む場合、Eステージ制御部70はAA1レジスタ70にアドレスを設定し、DDR2レジスタ70にデータを設定して、外部バスインターフェース制御部70に対してアクセス要求信号80を出力する。この時、リード/ライト信号80をライトにしておく。外部バスインターフェース制御部70は要求を受けると、Eステージ制御部70へアクセス要求受付信号80を返す。メモリ・アクセスが完了すると、外部バスインターフェース制御部70はEステージ制御部70へアクセス完了信号80を返す。

Eステージ制御部70からのアクセス要求とOPステージ制御部70からのアクセス要求が同時に起った場合、外部バスインターフェース制御部70はEステージ制御部70からのアクセス要求を優先的に受け付ける。

号とオペランド・バイパスの動作制御信号80のANDをとることによって、オペランド・バイパス機構70を動作させる場合は、オペランド・バイパス機構70の出力をそのままOPステージ制御部70へ伝え、動作させない場合は、常にバイパス不可とする。

データ演算部70がSDATA70中のデータを使用する前に、Eステージ制御部70からの書き込み要求により、SDATA70中のデータと同じアドレスに書き込みが行われた場合は、SCAM70によるオペランド・アドレス一致検出によりSDATA70中のデータの書き換えも行われ、データの一貫性が保たれるようになっている。

(5) 制御レジスタによる分岐予測機構およびオペランド・バイパス機構の制御

本発明のデータ処理装置では、分岐予測機構およびオペランド・バイパス機構の動作を制御するための制御レジスタを設けた。第3図に制御レジスタの構成を示す。

BBCレジスタのMビットによって、ブランチ予測機構の動作を制御する。M=0のときブランチ予

## 特開平1-216427 (16)

測機構は動作せず、M=1のとき動作する。このMビットは分岐予測機構の動作制御信号(160)となる。

BBCレジスタのBビットによって、オペランド・バイパス機構の動作を制御する。B=0のときオペランド・バイパス機構は動作せず、B=1のとき動作する。このBビットはオペランド・バイパス機構の動作制御信号(161)となる。

BBPレジスタのPビットによって、分岐履歴のクリアを行う。Pビットに1を書き込むことで分岐履歴はクリアされる。BBPレジスタは書き込み専用のレジスタである。BBPレジスタは値を保持するようにはなっており、BBPレジスタに書き込むという動作が直接何らかの動作を制御する信号になる。BBPレジスタを読み出したとしても、常に"0"が読み出されるだけである。このPビットへの書き込み信号は、分岐予測部のクリア信号(157)となる。

ここで述べた制御ビットは、別々の制御レジスタ中に存在してもよいし、共通の制御レジスタ中に存在してもよい。制御レジスタ中の適当な制御

と、命令をデコードする命令デコード機構とを備え、命令のデコードと、条件分岐命令に対して上記分岐予測機構の出力に従い第1の分岐処理を行うか上記第1の分岐処理を行わないかかどちらかの動作とを行う機構をもつ第1のパイプラインステージと、条件分岐命令に対する分岐条件に従い、第2の分岐処理を行うかまたは上記第2の分岐処理を行わないかどちらかの処理を行う第2のパイプラインステージと、上記第1の分岐処理動作を許可または禁止することを制御する分岐制御情報を記憶するレジスタとを備え、上記分岐制御情報を書き換える命令をもつことを特徴とするデータ処理装置。

(2)条件分岐命令の分岐履歴を保持する分岐履歴テーブルと、上記分岐履歴テーブルの出力値をもとに分岐予測を行い、上記分岐制御情報により上記分岐履歴テーブルの出力を「分岐しない」に固定することと、上記分岐履歴テーブルの書き換えを禁止することにより上記第1の分岐処理動作の禁止を行うことを特徴とする特許請求の範囲第1項

ビットをセット/リセットすることで、分岐予測機構やオペランド・バイパス機構の動作を制御することができる。

これらのレジスタへの値の設定はLDC命令(専用レジスタへのロード命令)によって行える。

また、LDCTX命令(コンテキストのロード命令)のようなタスクやプロセスの切り替えに用いられるような命令が実行されるとプログラムが実行される環境が変わるので、分岐履歴は全くあてにならないものとなる。そこで、このようなプログラムの実行される環境が変化するような命令が実行されると自動的に分岐予測部のクリア信号(157)が発生するようにする。

本実施例においては、分岐履歴のクリアは1度に全てのビットをクリアするようにしているが、分岐履歴の各ビットの値を任意に設定できるようにしてもよい。

本発明は次の(1)~(8)項の実施形態により実施できる。

(1)条件分岐命令の分岐予測を行う分岐予測機構

記載のデータ処理装置。

(3)条件分岐命令の分岐履歴を保持する分岐履歴テーブルと、上記分岐履歴テーブルの出力値をもとに分岐予測を行う分岐予測機構と、命令をデコードする命令デコード機構とを備え、命令のデコードと、条件分岐命令に対して上記分岐予測機構の出力に従い第1の分岐処理を行うか上記第1の分岐処理を行わないかかどちらかの動作とを行う機構をもつ第1のパイプラインステージと、条件分岐命令に対する分岐条件に従い、第2の分岐処理を行うかまたは上記第2の分岐処理を行わないかどちらかの処理を行う第2のパイプラインステージと、上記分岐履歴を書き換える命令と、を備えることを特徴とするデータ処理装置。

(4)上記分岐履歴の書き換えは分岐履歴を一律に「分岐しない」とする書き込みであることを特徴とする特許請求の範囲第3項記載のデータ処理装置。

(5)新しいプログラム実行環境を設定する動作をする命令に上記分岐履歴の書き換え動作を併せも

## 特開平1-216427 (17)

つことを特徴とする特許請求の範囲第4項記載のデータ処理装置。

(6) オペランドバイパス制御情報を保持する制御レジスタと、フェッチする第1のオペランドの第1のアドレス値とストアする第2のオペランドの第2のアドレス値とを比較するアドレス比較機構と、上記アドレス比較機構での比較結果と上記制御レジスタの出力とを入力としてその論理演算結果を出力する論理回路と、上記論理演算結果が「バイパス可能」を示すとき、メモリをバイパスして上記第2のオペランドの値を上記第1のオペランドとするバイパス機構と、を備え、上記オペランドバイパス制御情報を書き換える命令をもつことを特徴とするデータ処理装置。

(7) フェッチすべきオペランドのアドレスを保持するアドレスレジスタと、ストアするオペランドのアドレスを転送する第1のバスと、ストアするオペランドとフェッチするオペランドの両方を転送する第2のバスと、第1のバスに接続し、第1のバスの値を書き込むことが可能な第1記憶装置

と、第2のバスに接続し、第2のバス値を書き込むことが可能な第2記憶装置と、上記アドレスレジスタに保持された第1のアドレス値と上記第1のバスで転送されるストアするオペランドの第2のアドレス値とを比較する比較装置と、を備え、オペランドをストアするとき、上記第1のアドレス値と第2のアドレス値が一致する場合には、上記第2のアドレス値を上記第1の記憶装置に書き込み、上記第2のバスで転送されるオペランドを上記第2の記憶装置に書き込むことを特徴とするデータ処理装置。

(8) 上記ストア動作がワード境界をまたぐとき、上記第1の記憶装置に書き込むアドレス値はストア動作のバスサイクルごとのアドレスとし、上記第2の記憶装置に書き込むアドレス値はストア動作のバスサイクルごとのオペランドデータであること、を特徴とする特許請求の範囲第7項記載のデータ処理装置。

## 〔発明の効果〕

以上説明したように、本発明によれば、分岐予

測機構を用いることでかえって処理効率が落ちる場合には、制御レジスタへ値をセットすることにより、分岐予測機構を動作させないようにすることができ、処理効率を上げることができる。命令の流れを外郎アドレス・バスのモニタによって行う場合、制御レジスタへ値をセットすることにより分岐予測機構を動作させないようにして、命令の流れを追いやすくすることができる。

プログラムの走る状況が変化するなどして、分岐履歴があてにならなくなった場合は、制御レジスタへ値を書き込むことで分岐履歴をクリアでき、前に走っていたプログラムの悪影響を受けないようにすることができる。また、プログラムの実行状況を変化させるような命令（コンテキスト・スイッチ用の命令など）が実行された場合には、分岐履歴が自動的にクリアされる。

オペランド・フェッチを外郎アドレス・バスでモニタする場合などは、制御レジスタへ値をセットすることで、オペランド・バイパス動作をさせないようにすることができ、バスによるチップ動

作のモニタが容易になる。

## 4. 図面の簡単な説明

第1図は本発明のデータ処理装置の分岐予測テーブル詳細図、第2図は本発明のデータ処理装置のバスインターフェイス制御部およびその周辺を示すブロック図、第3図は本発明の制御レジスタの構成図、第4図は本発明のデータ処理装置の分岐命令処理回路の図、第5図は本発明のデータ処理装置の全体ブロック図、第6図は本発明のデータ処理装置のパイプラインステージ構成図、第7図は従来のデータ処理装置のパイプラインステージ構成図、第8図は従来のデータ処理装置での分岐命令処理の様子を示す図である。

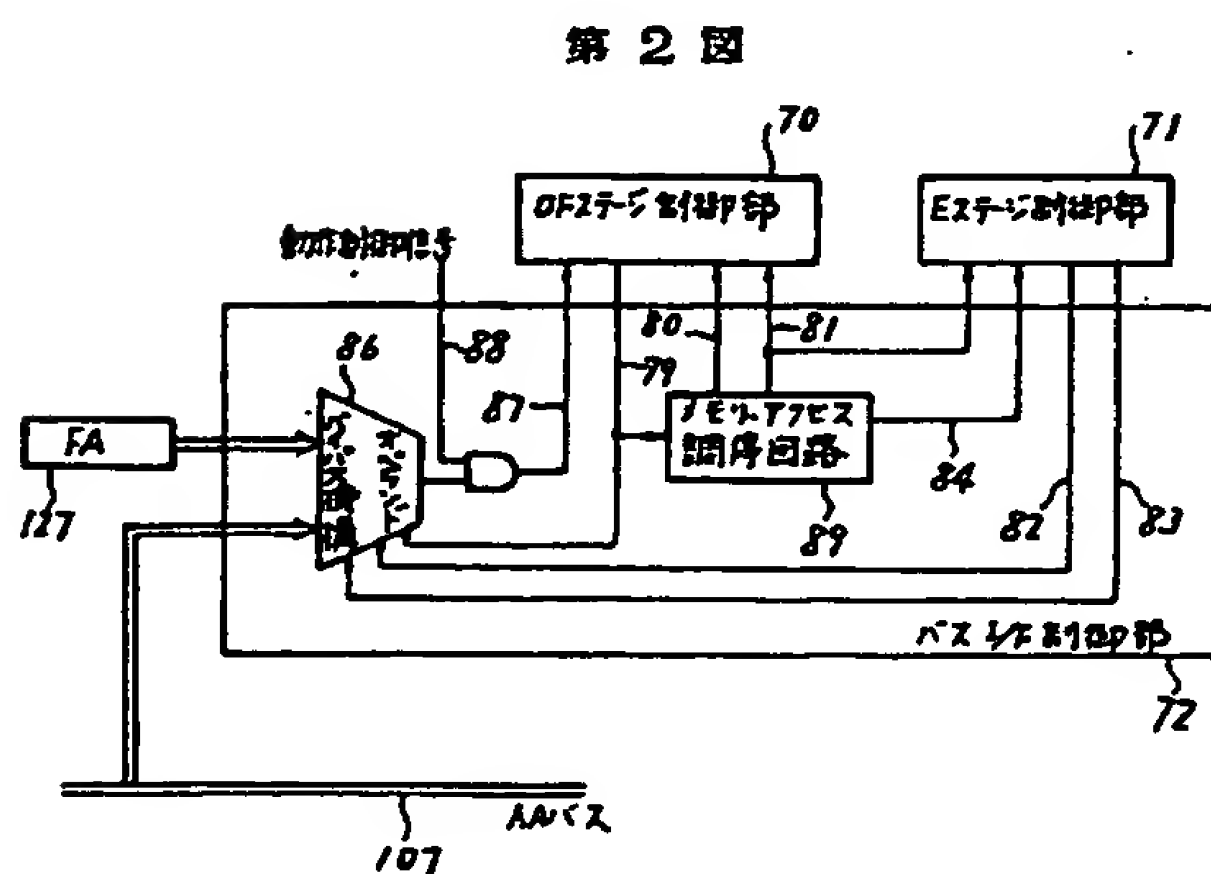
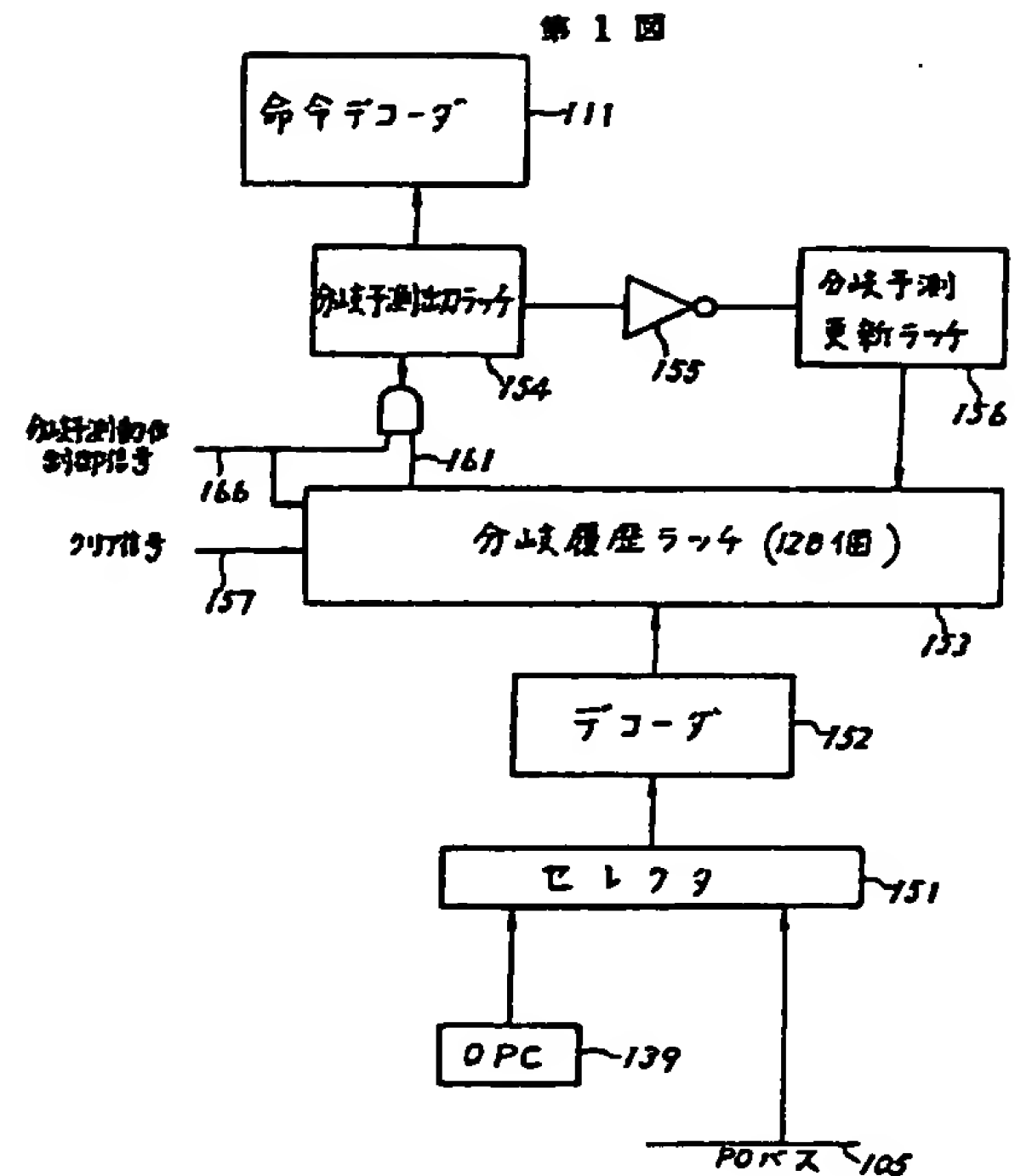
図において、(105)はPOバス、(111)は命令デコード、(139)はOPCレジスタ、(151)はPOバスとOPCレジスタのセレクタ、(152)はデコード、(153)は分岐予測ラッチ、(154)は分岐予測出力ラッチ、(155)は予測反転回路、(156)は分岐予測更新ラッチ、(157)は分岐履歴クリア信号、(160)は分岐予測動作制御信号、(161)は分岐予測信号である。

図において、70はOPステージ制御部、71はEステージ制御部、72は外部バスインターフェイス制御部、73はOPステージ制御部からのアクセス要求信号、74はOPステージ制御部へのアクセス要求受付信号、75はアクセス完了信号、76はEステージ制御部からのアクセス要求信号、77はEステージ制御部からのアクセス要求がリードかライトかを示す信号、78はEステージ制御部へのアクセス要求受付信号、79はオペランド・バイパス機構、80はオペランド・バイパス機構から出力されるバイパス可能信号、81はオペランド・バイパス機構の動作を制御する信号、82はメモリアクセス制御回路である。

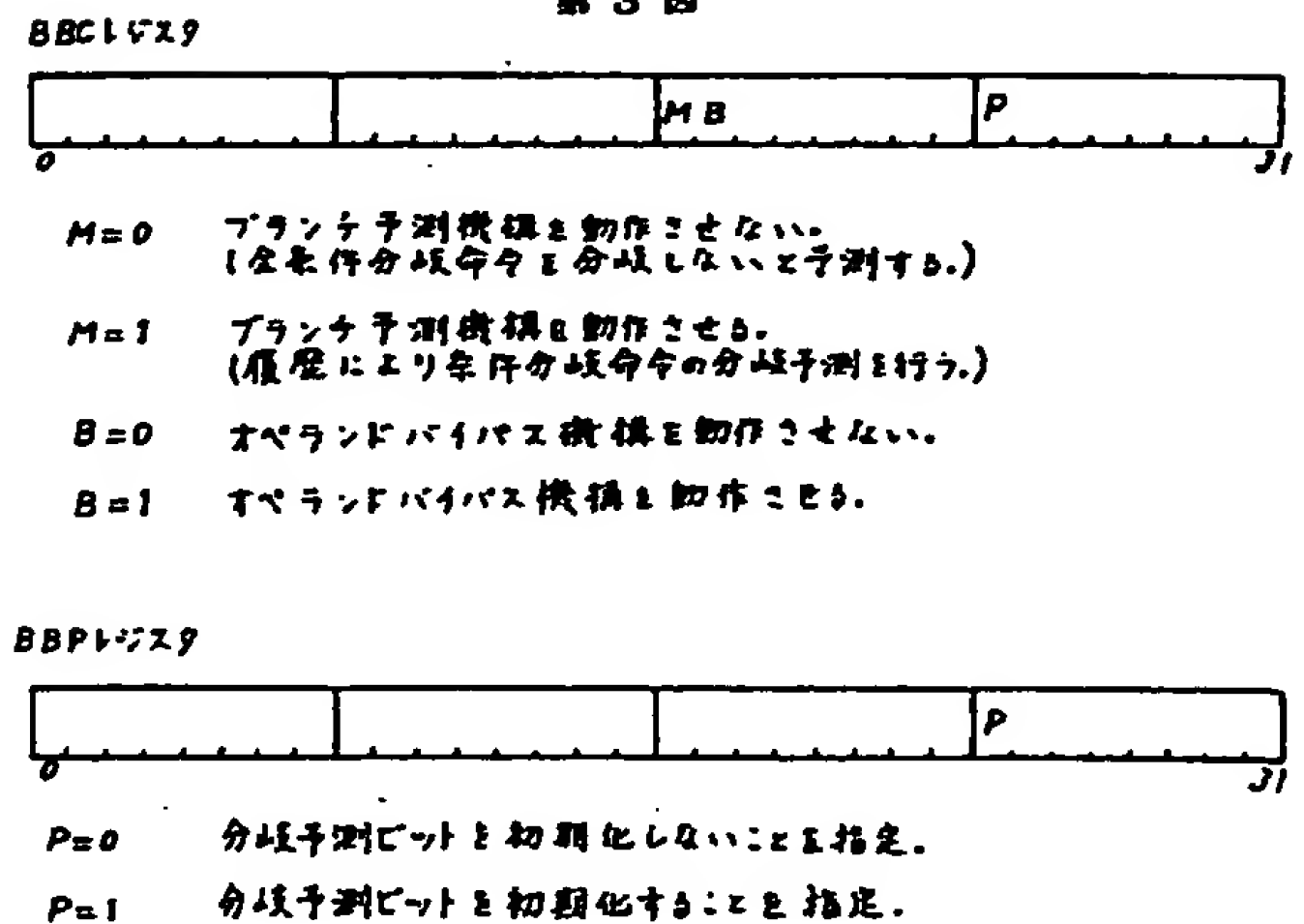
尚、各図中同一符号は同一または相当部分を示す。

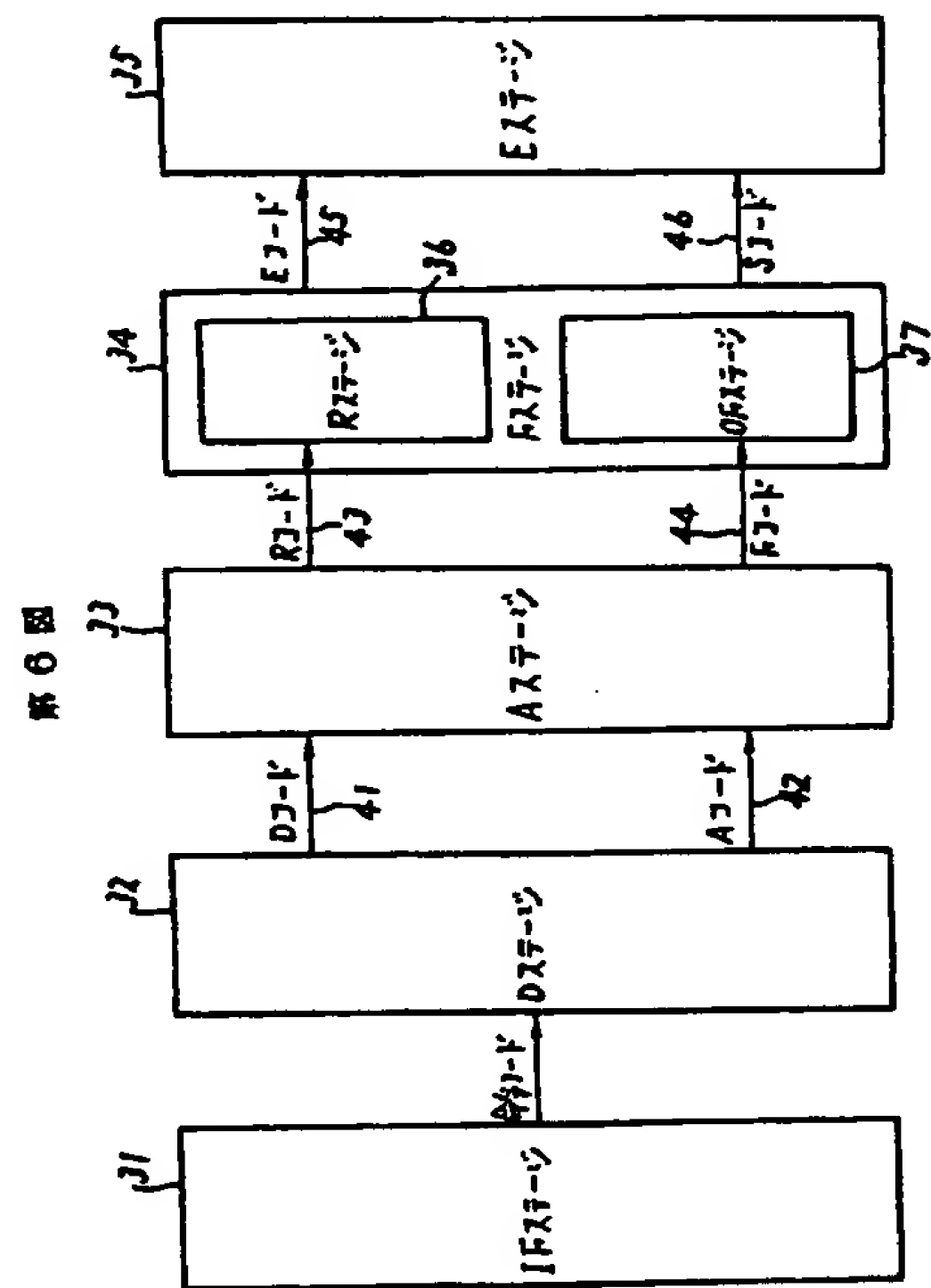
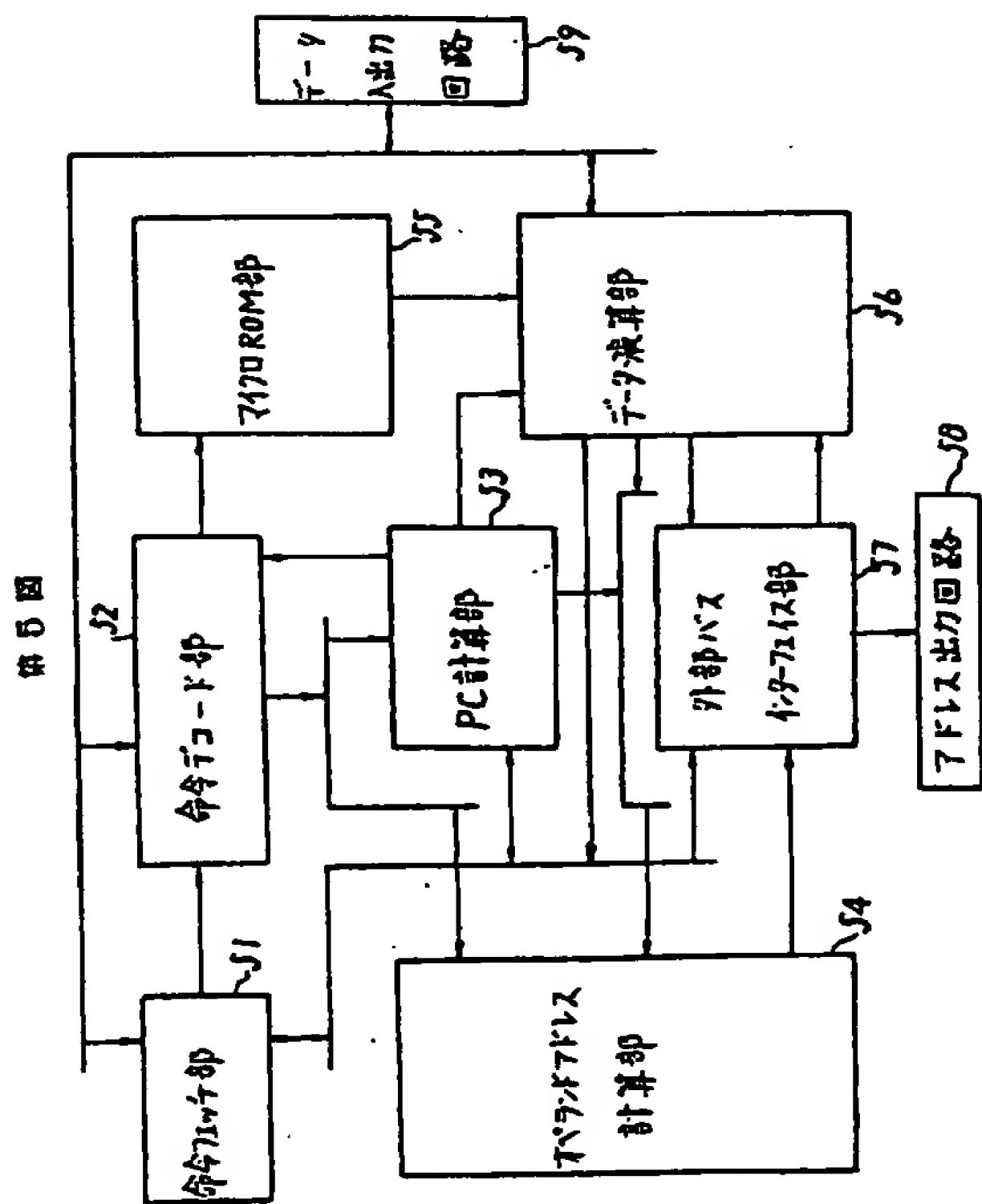
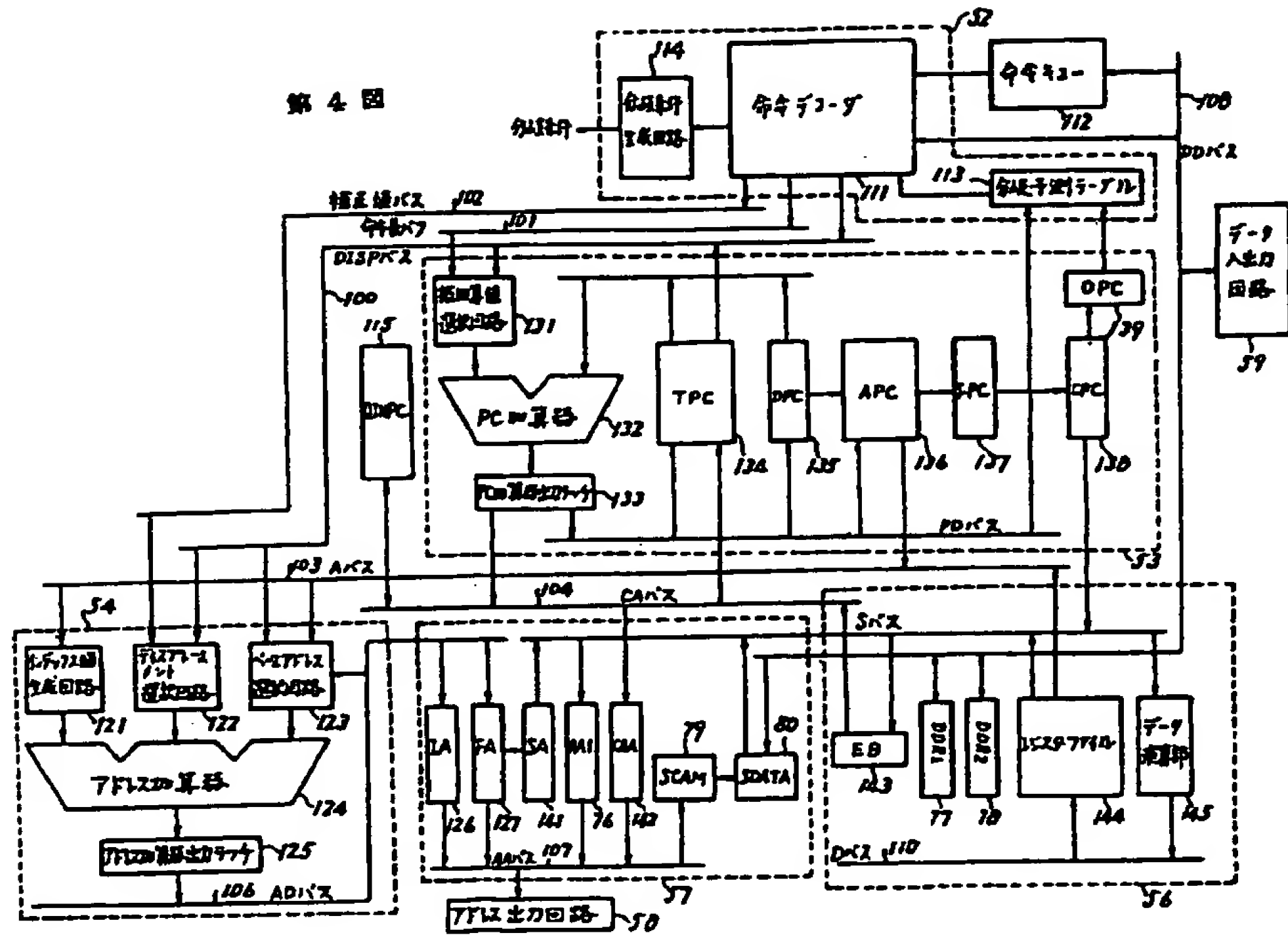
代理人 大 昔 増 雄

特開平 1-216427 (18)



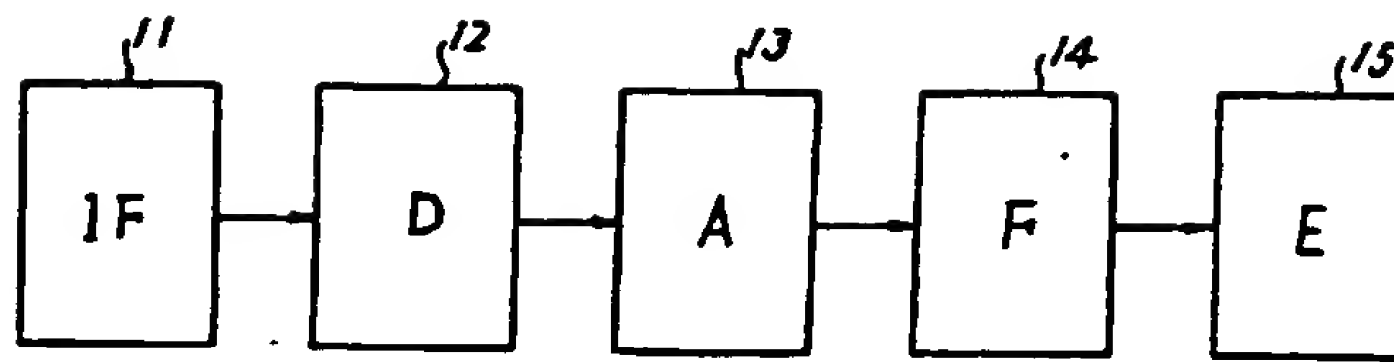
第 3 図



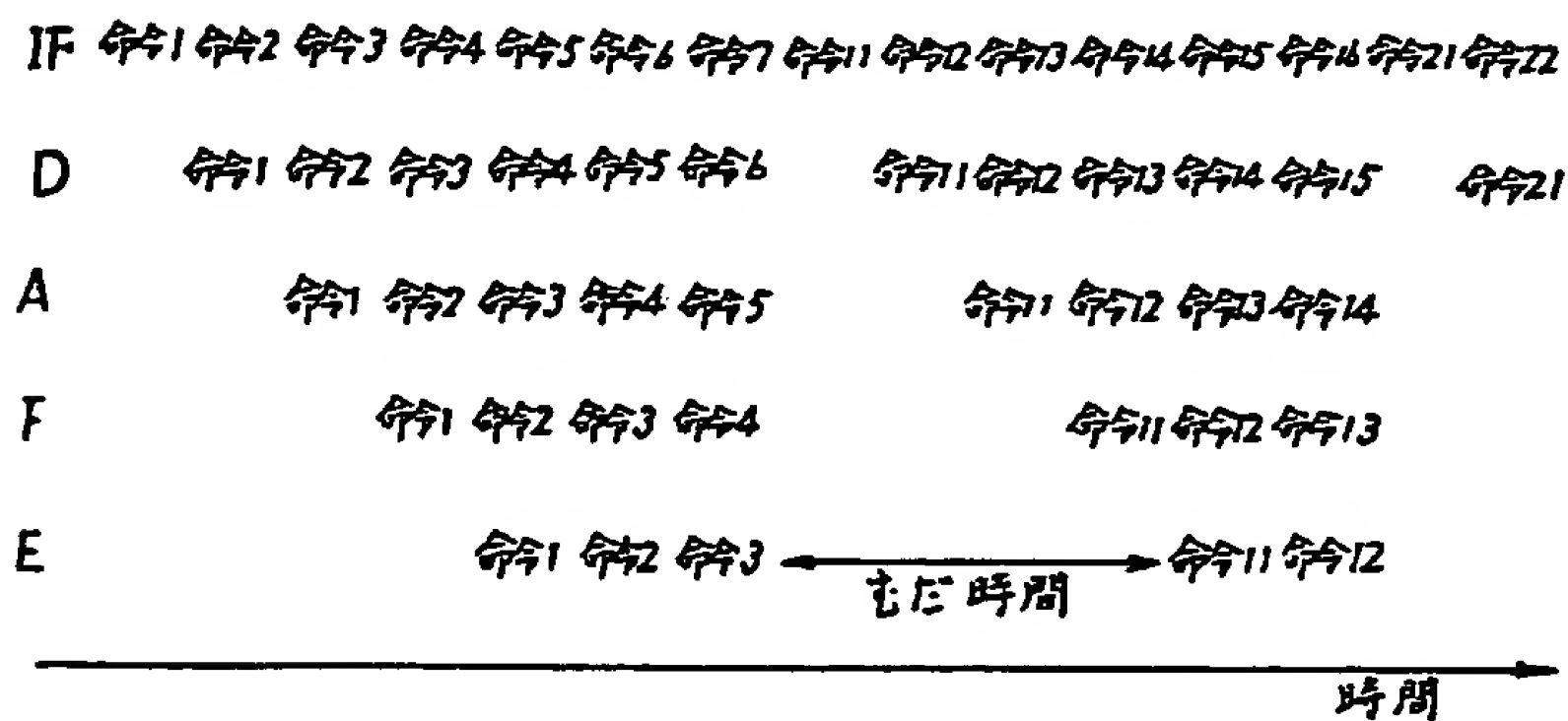


特開平1-216427 (20)

第7図



第8図



手続補正書(自発)

昭和 83 年 7 月 1 日

特許庁長官殿

1. 事件の表示 特願昭 63-41086 号

2. 発明の名称  
データ処理装置

3. 補正をする者

事件との関係 特許出願人  
住 所 東京都千代田区丸の内二丁目2番3号  
名 称 (601) 三菱電機株式会社  
代表者 志 岐 守 哉

4. 代 理 人

住 所 東京都千代田区丸の内二丁目2番3号  
三菱電機株式会社内  
氏 名 (7375) 弁理士 大 岩 増 雄  
(連絡先 03(213)3421 特許部)

5. 補正の対象

明細書の発明の詳細な説明の欄及図面

6. 補正の内容

(1) 図面中、第1図を別紙のとおり訂正する。

(2) 明細書をつぎのとおり訂正する。

ページ	行	訂 正 前	訂 正 後
6	10~11	分岐命令をIFステージ04が処理する	分岐先命令をIFステージ04が取り込む
6	14	場合に	場合の
6	17	命令5、命令7	命令5、命令6、命令7
16	7~8	DDR1レジスタ	DDB1レジスタの
29	18	(Fステージ04も	(Fステージ04)も
30	18	ステップ行われる	ステップで行われる
62	8	動作とを	動作を
61	20	特許請求の範囲	上記
62	17	・	・
63	1	・	・
64	17	・	・



特開平 1-216427 (21)

第 1 図

